Aufgabe: Mit den Temperatursensor TMP36 soll die Temperatur ausgelesen und mit dem serial-monitor angezeigt werden.

Material:

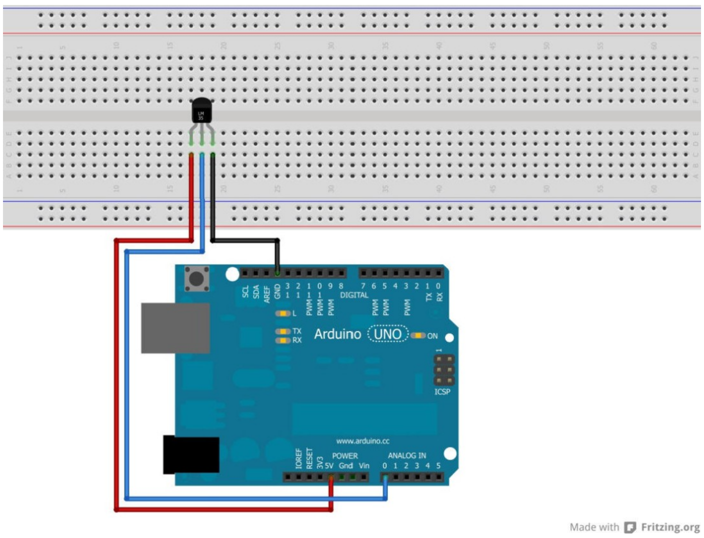
- Arduino-Board
- Temperatursensor TMP36
- wahlweise externe Stromversorgung
- Breadboard
- Steckkabel

In dieser Anleitung lernst du:

- Sensorwerte auslesen und auf dem Serial Monitor anzeigen lassen
- „Map“-Befehl
- Temperatursensor anschließen

Der Temperatursensor TMP36 hat drei Anschlüsse. Beim Blick auf die flache Seite des Sensors: links 5V, rechts GND und in der Mitte der Pin für das Temperatursignal. Auf diesem Pin gibt der Sensor eine Spannung zwischen 0 und 2,0 Volt aus. Wobei 0V -50°C entsprechen und der Wert 2,0V entspricht 150°C. Laut Hersteller ist der Sensor zwischen -40°C und +125°C einigermaßen genau (±2°C). Die Spannung dieses Pins muss vom Mikrocontroller-Board ausgelesen und in einen Temperaturwert umgerechnet werden.
- ACHTUNG: Wenn der Sensor falsch angeschlossen wird, brennt er durch!
- ACHTUNG: Bei schwankender Spannung am USB-Anschluss des Arduino Mikrocontrollerboards können die Messwerte verfälscht werden. In dem Fall empfiehlt es sich, eine externe Spannungsversorgung zu verwenden. Die Messwerte sollten dann auf einem LCD-Display angezeigt werden.

Aufbau:



Für diesen Sketch wird der "map" Befehl benötigt. Dieser Befehl befindet sich in der Zeile: "temperatur= map(analogRead(TMP36), 0, 410, -50, 150);"
Anhand der allgemeinen Schreibweise „map (a, b, c, d, e)“ lässt sich die Funktion besser beschreiben. Ein Wert "a" (beispielsweise ein Messwert) wird in einem bestimmten Zahlenbereich zwischen den zwei Werten (b) und (c) erwartet. Der "map" Befehl wandelt dann den Wert "a" in einen anderen Wert um, der dem Zahlenbereich zwischen "d" und "e" entspricht.
In unserem Sketch passiert dabei folgendes:
Der Temperatursensor TMP36 gibt an dem mittleren Pin den Messwert für die Temperatur in Form einer Spannung zwischen 0V und 2V aus. Dieser Spannungsbereich entspricht dem messbaren Temperaturbereich von -50°C bis +150°C. Am analogen Eingangspin des Arduino Mikrocontrollerboards wird dieser Spannungsbereich mit Hilfe des Befehls "analogRead(TMP36)" als Zahlenwert zwischen 0 und 410 erkannt. Dieser Wert des Temperatursensors wird zunächst ausgelesen und unter der Variablen "sensorwert" gespeichert.
Der "map" Befehl wird nun verwendet um diesen Zahlenwert zwischen 0 und 410 wieder in einen Temperaturwert zwischen -50°C und +150°C umzuwandeln.
Temperatur = map(sensorwert, 0, 410, -50, 150);
temperatur = map (a , b , c , d , e)
a= umzuwandelnde Zahl
b= minimum Messbereich
c= maximum Messbereich
d= minimum Ausgabewert
e= maximum Ausgabewert

Nach der Umwandlung des analogen Messwertes in einen Temperaturwert, wird dieser mit dem Befehl "**Serial**.print(temperatur);" an den seriellen Monitor gesendet und kann dann am PC abgelesen werden.

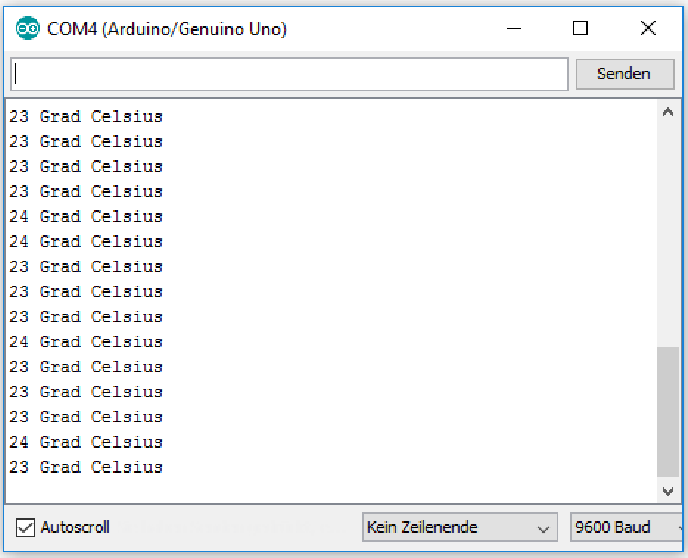
Sketch zur Temperaturmessung:

```
int TMP36 = A0;              //Der Sensor soll am analogen Pin A0 angeschlossen
                             werden. Wir nennen den Pin ab jetzt "TMP36"
int sensorwert;
int temperatur = 0;          //Unter der Variablen "temperatur" wird später der
                             Temperaturwert abgespeichert.
int t=500;                   //Der Wert für „t" gibt im Code die zeitlichen
                             Abstände zwischen den einzelnen Messungen vor.

void setup()
{
Serial.begin(9600);          //Im Setup beginnt die serielle Kommunikation,
                             damit die Temperatur an den serial monitor
                             übertragen wird. Über die serielle Kommunikation
                             sendet das Board die Messwerte an den Computer. In
                             der Arduino-Software kann man unter „Werkzeuge"
                             den „Seriellen Monitor" starten um die Messwerte
                             zu sehen.
}

void loop()
{
sensorwert=analogRead(TMP36); //Auslesen des Sensorwertes.
temperatur= map(sensorwert, 0, 410, -50, 150); //Umwandeln des Sensorwertes mit
                                                 Hilfe des "map" Befehls.
delay(t);                    // Nach jeder Messung ist je eine kleine Pause mit
                             der Dauer „t" in Millisekunden.
Serial.print(temperatur);    //Nun wird der Wert „temperatur" über die serielle
                             Kommunikation an den PC gesendet. Durch öffnen des
                             seriellen Monitors in der Arduino-Software kann
                             die Temperatur abgelesen werden.
Serial.println(" Grad Celsius"); // Im seriellen Monitor wird hinter der
                                    Temperatur die Einheit eingeblendet.
}
```

Nach dem öffnen des seriellen Monitors sollte das Ergebnis so aussehen:



Erweiterung des Programms:

Sobald die Temperatur von 30°C erreicht ist, soll ein Warnsignal ertönen. Dazu wird ein Piezo-Lautsprecher mit dem „+“ Pol an Pin5 des Arduino Mikrocontrollerboards angeschlossen. Der andere Pin des Lautsprechers wird mit GND verbunden.

```
int TMP36 = A0;
int sensorwert;
int temperatur = 0;
int t=500;
int piezo=5;                  //Das Wort „piezo“ steht jetzt für die Zahl
                              5, also wird an Pin5 der Speaker
                              angeschlossen.

void setup()
{
Serial.begin(9600);
pinMode (piezo, OUTPUT);      //Der Pin für den Piezo-Lautsprecher wird
                              als Ausgang definiert, da hier um zu piepsen
                              eine Spannung benötigt wird.
}

void loop()
{
sensorwert=analogRead(TMP36);
temperatur= map(sensorwert, 0, 410, -50, 150);
delay(t);
Serial.print(temperatur);
Serial.println(" Grad Celsius");

if (temperatur>=30)           //Es wird eine IF-Bedingung erstellt: Wenn
                              der Wert für die Temperatur über oder gleich
                              30 ist, dann…
{
digitalWrite(piezo,HIGH);     //…fange an zu piepsen.
}
else                          //Und wenn das nicht so ist…
{
digitalWrite(piezo,LOW);      //…dann sein leise.
}
}
```